用于边缘计算设备的果树挂果量轻量化估测模型

夏　雪[1]，柴秀娟[1]，张　凝[1*]，周　硕[1]，孙琦鑫[1]，孙　坦[2*]

（1. 中国农业科学院农业信息研究所/农业农村部农业大数据重点实验室，北京 100081；2. 中国农业科学院，北京 100081）

摘　要：［目的/意义］挂果量是果树栽培管理的重要指标。传统人力抽样估测果树挂果量的方法不仅耗时费力，而且容易产生较大误差。本研究提出一种用于边缘计算设备的轻量化模型，实现视频中树上柑橘挂果量的自动估测。［方法］该模型采用CSPDarkNet53+PAFPN结构作为特征提取网络，实现更快的推理速度和更低的模型复杂度，在果实跟踪过程中引入Byte算法改进FairMOT的数据关联策略，对视频中的柑橘进行预测跟踪，以提升挂果量估测准确性。［结果和讨论］在边缘计算设备NVIDIA Jetson AGX上进行模型性能测试结果表明，本研究所建模型对柑橘挂果量的平均估测精度（Average Estimating Precision，AEP）和处理速度（Frames Per Second，FPS）分别达到91.61%和14.76，模型估测值与人工测得真实值的决定系数R^2为0.9858，均方根误差（Root Mean Square Error，RMSE）为4.1713，模型参数量、计算量（Floating Point Operations，FLOPs）和模型大小分别为5.01 M、36.44 G和70.20 MB，展现出较对比模型更优的挂果量估测性能和更低的模型复杂度。［结论］试验结果证明了本研究所建模型在边缘计算设备上对柑橘挂果量估测的有效性，基于算法模型研发的果园挂果量远程监测系统可满足用于果园移动平台行进状态下的果树挂果量估测需求。本研究可为果园生产力自动监测分析提供技术支持。

关键词：精准园艺；挂果量估测；边缘计算；深度学习；多目标跟踪；轻量化模型

中图分类号：TP181；S225.93　　**文献标志码：**A　　**文章编号：**SA202305004

引用格式：夏雪，柴秀娟，张凝，周硕，孙琦鑫，孙坦．用于边缘计算设备的果树挂果量轻量化估测模型[J]. 智慧农业(中英文)，2023，5(2)：1-12.

XIA Xue, CHAI Xiujuan, ZHANG Ning, ZHOU Shuo, SUN Qixin, SUN Tan. A lightweight fruit load estimation model for edge computing equipment[J]. Smart Agriculture, 2023, 5(2): 1-12.

1　引　言

果树挂果量是评价果园生产力的重要指标，可使农户更精确地掌握果园生产情况，合理安排栽培管理、果实采收、储存和销售等活动[1]。估测树上挂果量的传统方法主要依赖人工完成，如随机抽取固定百分比（5%或10%）的果树并进行果实计数，据此推断整个果园的果实数量。然而，这种长时间抽样和户外工作不仅费时费力，而且容易因大脑疲劳或其他干扰而产生计数错误。因此，自动估测果树挂果量对果园生产至关重要。机器视觉作为人工智能应用的重要领域，因其低成本、高效率的优点，已广泛应用于智慧果园相关研究。

早期的果树挂果量估测研究多关注于传统基于手工特征的图像处理方法，如纹理特征[2,3]、颜色特征[4,5]、形状特征[6,7]等。近年来，深度学习技术在果树挂果量估测方面得到广泛研究。Sa等[8]在研究中较早尝试了利用深度学习的果实检测来估测果树挂果量。Chen等[9]通过全卷积网络提取候

收稿日期：2023-05-11

基金项目：国家重点研发计划项目（2022YFD2002205）；中央级公益性科研院所基本科研业务费专项（Y2022QC17，JBYW-AII-2022-05，JBYW-AII-2022-26）；中国农业科学院科技创新工程（CAAS-ASTIP-2021-AII-08，CAAS-ASTIP-2023-AII）

作者简介：夏　雪，助理研究员，研究方向为果树表型研究与应用。E-mail：xiaxue@caas.cn

*通信作者：1. 张　凝，博士，副研究员，研究方向为作物表型、定量遥感。E-mail：zhangning@caas.cn；
2. 孙　坦，博士，研究馆员，研究方向为数字信息描述与组织、智慧农业。E-mail：suntan@caas.cn

选blob，并使用与卷积神经网络相关的回归模型来实现每个blob区域果实数量的估测。Bargoti和Underwood[10]提出了一种基于Faster RCNN和迁移学习的架构来估计挂果量的方法。Häni等[11]研发了一种结合U-Net与Faster RCNN的端到端系统，可从苹果簇中估计果实数量，准确率达到0.978。李志军等[12]利用YOLOv5检测图像中的苹果，并结合产量拟合网络实现了果树产量的预测。Kestur等[13]专门设计了一种MangoNet模型，实现芒果挂果量估测。

然而，现有方法大多基于静态图像来实现挂果量估测，无法完成对视频中果树挂果量的动态估测任务。为解决上述难题，基于检测的跟踪（Tracking by Detection，TBD）方法开始被一些学者引入到估测视频中果树挂果量的研究任务中。高芳芳等[14]提出基于YOLOv4-tiny和卡尔曼滤波跟踪的方法，实现了视频中富士苹果的检测和计数。类似地，Wang等[15]利用卡尔曼滤波器对芒果目标进行运动跟踪，实现了视频中芒果数量的估测。TBD类方法采用了两阶段策略，即先用检测模型完成目标检测，然后将输出的检测结果送入另一个模型，利用数据关联算法完成多目标跟踪，因此会导致算法效率偏低、处理速度较慢[16,17]等问题，而且算法大多只能在高性能图形工作站上运行，在实际果园生产中难以适用于硬件资源有限的边缘计算设备。

随着多目标跟踪技术的快速发展，基于联合检测和嵌入（Joint Detection and Embedding，JDE）的方法逐渐成为主流。JDE类方法采用端对端的单阶段策略，将检测和跟踪整合到一个框架中同时实现目标检测任务和身份重识别（ReID）任务，避免了多阶段处理的算法复杂性[18]。Zhang等[19]基于CenterNet和JDE提出了FairMOT框架，在目标跟踪阶段同时考虑目标位置及其ReID特征，通过共享大量计算来减少模型推理时间。Zhang等[20]提出的ByteTrack算法，在考虑高置信度检测框的同时，也将低置信度检测框加到匹配过程中来挖掘更多真实目标，带来了推理精度与速度两方面的提升。然而，该方法的关联匹配仅采用了运动估计策略，并未引入身份重识别策略来计算相似度[21]。

为实现果园工况条件下果树挂果量的自动估测，本研究以树上柑橘为对象，研究用于边缘计算设备的树上柑橘挂果量估测模型。研究主要贡献包括：（1）采用具有轻量化特点的CSPDarkNet53+PAFPN结构作为模型特征提取网络，保证特征表征能力的同时实现更快的推理速度和更低的模型复杂度；（2）在果实目标跟踪阶段，引入Byte算法改进FairMOT的数据关联策略，设计果实多目标跟踪模块，对柑橘果实进行预测跟踪，提高挂果量估测准确性；（3）研发果园挂果量远程监测系统，将算法模型嵌入监测系统，实现果树挂果量的自动估测。

2　试验数据采集与处理

2.1　数据采集

试验果园位于广西壮族自治区南宁市江南区那廊村（108°06′E，22°79′N），以此处种植的沃柑为研究对象，于2019年11月下旬采集了柑橘视频图像数据。柑橘园种植环境如图1（a）所示。研究人员手持智能手机沿果树树行移动进行拍摄，手机摄像头距离树行0.5～1.0 m，试验数据的采集在晴朗及多云天气下完成，采集时段为8：00—18：00，确保获取的数据涵盖不同的光照情况。

采集果园中不同行的果树视频40个，保存为*.mp4格式，分辨率像素为1080×1920，视频帧率为24帧/s。采集柑橘视频的图像样例如图1（b）所示。



（a）柑橘种植环境



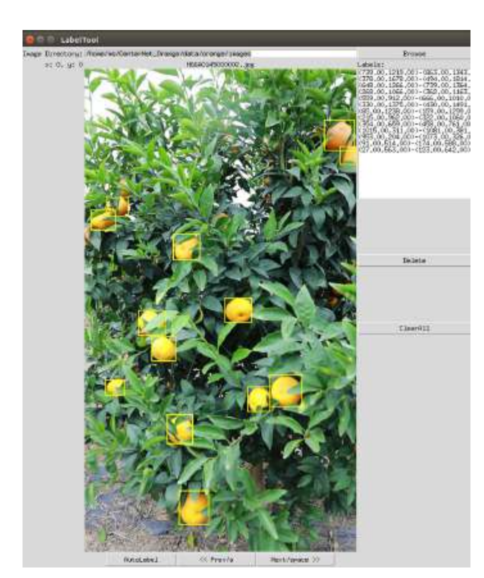
（b）柑橘视频图像样例

图1　广西壮族自治区南宁市江南区柑橘果园

Fig. 1　Citrus orchard in Jiangnan district，Nanning city，Guangxi Zhuang autonomous region

2.2 数据集构建

根据试验需求，从采集的果树视频中随机选择30个视频用于模型训练，其余10个视频作为挂果量测试视频。为了能够标注出视频中果实的真实数量（Ground truth），从视频数据中抽取图片数据，间隔为每30帧抽取1幅图像，共获得用于模型训练的柑橘树图像2846幅。对图像序列中的果实位置和重复果实进行标注。首先，使用自行开发的坐标标注工具（图2（a））进行果实最小外接矩形框的标注，记录每张图像中标注果实边界框的四个角点坐标；其次，使用自行开发的果实配对标注工具标注相邻图像中的重复果实（图2（b）），记录果实的序号，其中相同果实拥有相同的序号，最终形成标准可用的数据集。



(a)果实坐标标注工具



(b)果实配对标注工具

图2 果实标注工具

Fig. 2 Fruit labeling tools

3 轻量化果树挂果量估测模型

3.1 算法总体思路

在柑橘园复杂场景下，枝叶和果实之间容易产生遮挡，这给树上柑橘挂果量的估测带来一定挑战。现有目标跟踪方法大多采用基于锚框的目标检测[22]，如果目标实际中心位置与其检测产生的锚框中心有偏差，那么在锚点提取的身份特征有可能无法与目标对象中心对齐，导致跟踪精度降低。为了缓解对齐的问题，无锚框检测的多目标跟踪模型[19]被设计出来，不仅可以减少锚框对身份重识别（ReID）的影响，同时也能优化目标检测的推理速度。

基于上述分析，本研究借鉴FairMOT的思路，采用目标检测分支与身份嵌入分支的并行结构来设计果树挂果量估测模型。利用无锚框策略来生成检测目标的相应边界框，能够尽可能避免锚框策略产生的歧义问题，有利于身份嵌入特征更好对齐检测目标的中心位置。

本研究所提算法的总体思路如下：（1）检测视频图像中的果实并提取果实ReID特征。采用轻量化的网络结构CSPDarknet53+PAFPN用于提取特征图，之后特征图会被分别送到两个并行分支，一个分支用来预测目标位置信息，另一个分支用来识别目标ReID特征；（2）果实跟踪与挂果量计算。在果实跟踪阶段引入Byte算法[20]，结合帧间果实目标位置信息及其身份特征进行果实运动轨迹预测和身份特征相似度匹配，对树上柑橘目标进行时序联结，统计果实跟踪持续帧数大于5帧的果实ID数量作为视频中树上柑橘的挂果量，最后输出果树挂果量估测结果。模型的算法处理流程如图3所示。

3.2 柑橘果实检测与ReID特征提取

本研究所建模型中果实检测与ReID特征提取网络的结构如图4所示。包括主干网络（Backbone）部分、Neck部分和预测头分支部分。

CSPDarknet53是以Darknet53[23]为基础，借鉴跨阶段局部网络CSPNet（Cross Stage Partial Network）的思想[24]形成的新网络，解决了网络反向优化引起的梯度信息冗余问题，有效减少了网络参

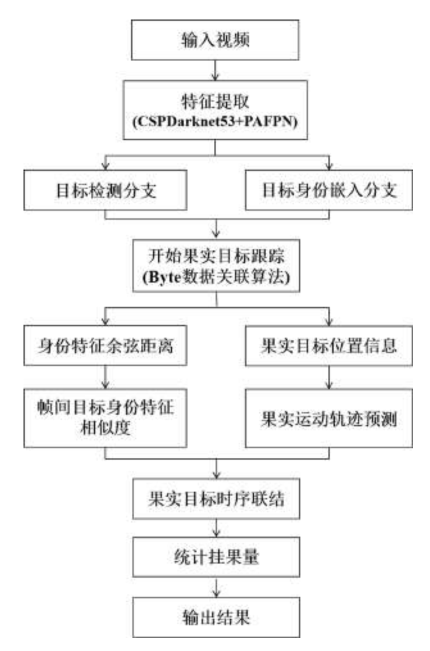


图3　果树挂果量估测算法处理流程图

Fig. 3 Flow chart of algorithm processing for fruit load estimation

数和计算量[25,26]。轻量化网络CSPDarknet53对硬件计算资源的消耗较少，适合图像处理资源相对有限的边缘计算设备[27]。因此，采用CSPDarknet53作为模型的主干网络来实现特征提取。

PAFPN由特征金字塔（Feature Pyramid Network，FPN）[28]和路径聚合网络（Path Aggregation Network，PAN）[29]组成，通过构建多尺寸的特征金字塔结构，将骨干网络提取的深层特征与浅层特征逐元素融合，充分利用低层特征的高分辨率和高层特征的高语义信息，以获取更加丰富的特征信息[30]。因此，在模型骨干网络后的Neck部分接入PAFPN特征金字塔网络，利用低层和高层特征之间的跳跃连接实现特征有效融合。

经过融合后的特征会分别送入果实检测分支（Detection branch）和果实身份嵌入（Identity embedding branch）分支。果实检测分支包括三个预测头，每个预测头都对PAFPN网络输出的特征图进行3×3卷积和1×1卷积，分别得到估计柑橘的中心热图（Keypoint Heat Map），中心偏移量（Local Offset）和边界框尺寸（Object Size），如图5所示。果实身份嵌入分支负责区分不同果实身份重识别（ReID）特征，网络设计中将卷积核的通道数由128个调整为64个，以减少推理计算时间，降低训练过拟合的风险，利用卷积核提取输入特征图上的ReID特征，产生身份嵌入图$E \in R^{64 \times W \times H}$，其中$W$和$H$分别表示特征图的宽和高，目标中心（$x$，$y$）处提取的身份（ReID）特征为$E_{x,y} \in R^{64}$。

3.3 融合Byte数据关联算法的柑橘果实跟踪

数据关联是多目标跟踪的关键环节。现有目标跟踪模型的数据关联大多是基于高分检测框（检测框得分高于设定阈值）匹配的策略，未考虑低分检测框（检测框得分低于设定阈值）目标。获得低分的检测框目标往往是由于遮挡或运动模糊等因素造成，若直接将低分检测框舍弃，容易使一些遮挡严重的跟踪目标丢失轨迹，引起频繁身份（ID）切换问题。倘若果实跟踪过程中同时考虑高分检测框和低分检测框来关联轨迹，将低分检测框利用起来，有利于提高目标轨迹连贯性。因此，本研究引入ByteTrack模型[20]中的Byte数据关联算法，同时考虑ReID特征和位置关联来设计柑橘果实跟踪模块。将柑橘果实检测结果作为跟踪模块的输入，对检测

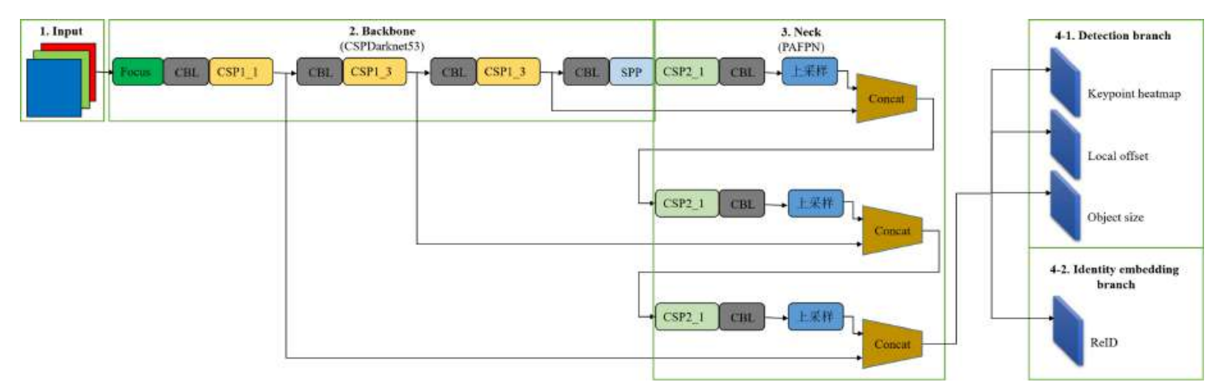
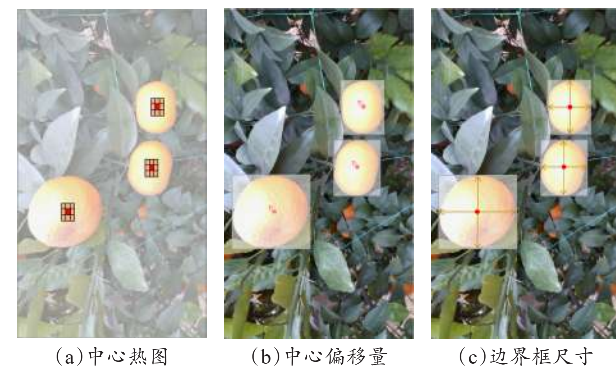


图4　柑橘果实检测与ReID特征提取网络结构图

Fig. 4 Structure diagram of citrus fruit detection and ReID feature extraction network



(a)中心热图　(b)中心偏移量　(c)边界框尺寸

图5　柑橘果实检测分支得到的中心热图、中心偏移量和边界框尺寸示意图

Fig. 5 Schematics of the keypoint heat map, local offset and object size from the citrus fruit detection branch

结果进行高分检测框和低分检测框的区分，并将检测框位置信息与ReID信息结合，形成融合特征，采用Byte数据关联匹配策略，获得连续视频帧中果实目标的轨迹，最后输出柑橘果实跟踪结果。果实跟踪的流程如图6所示。

柑橘果实跟踪具体步骤如下：

（1）设V为输入果树视频，Det为果实检测器，KF为卡尔曼滤波器。T为视频的轨迹栈，其中每条轨迹包括果实目标的检测框信息和身份信息。设定三个阈值T_{high}，T_{low}和e，其中T_{high}和T_{low}为目标检测置信度阈值，e为目标跟踪置信度阈值。

（2）对于所有由检测器Det产生的检测框和置信度，检测置信度高于阈值T_{high}的归类为D_{high}，D_{high}包含高于阈值T_{high}的检测框的位置及其ReID特征信息；检测置信度高于阈值T_{low}的归类为D_{low}，包含高于阈值T_{low}的检测框的位置信息及其ReID特征信息。

（3）对于轨迹栈T中的轨迹，利用卡尔曼滤波器KF预测其在当前帧中的坐标。

（4）将D_{high}与轨迹栈T中所有的轨迹进行第一次关联，借助检测框之间的交并比（Intersection Over Union，IOU）和身份特征，获得D_{high}检测框与滤波预测的检测框之间的相似度，利用匈牙利算法完成匹配。未匹配成功的检测框，存入D_{remain}堆栈，未匹配成功的轨迹，存入T_{remain}堆栈。

（5）将低置信度的检测框D_{low}与轨迹T_{remain}进行第二次关联，其匹配方法与第一次匹配的方法相同。对于未匹配成功的轨迹存入$T_{re\text{-}remain}$，同时，直接删除未匹配成功的低置信度检测框，不存入D_{remain}堆栈。

（6）对于$T_{re\text{-}remain}$中的轨迹，视为暂时丢失了目标，但$T_{re\text{-}remain}$中的轨迹仍会保留在轨迹栈T中。如果$T_{re\text{-}remain}$中轨迹后期匹配成功或是存在超过30帧，

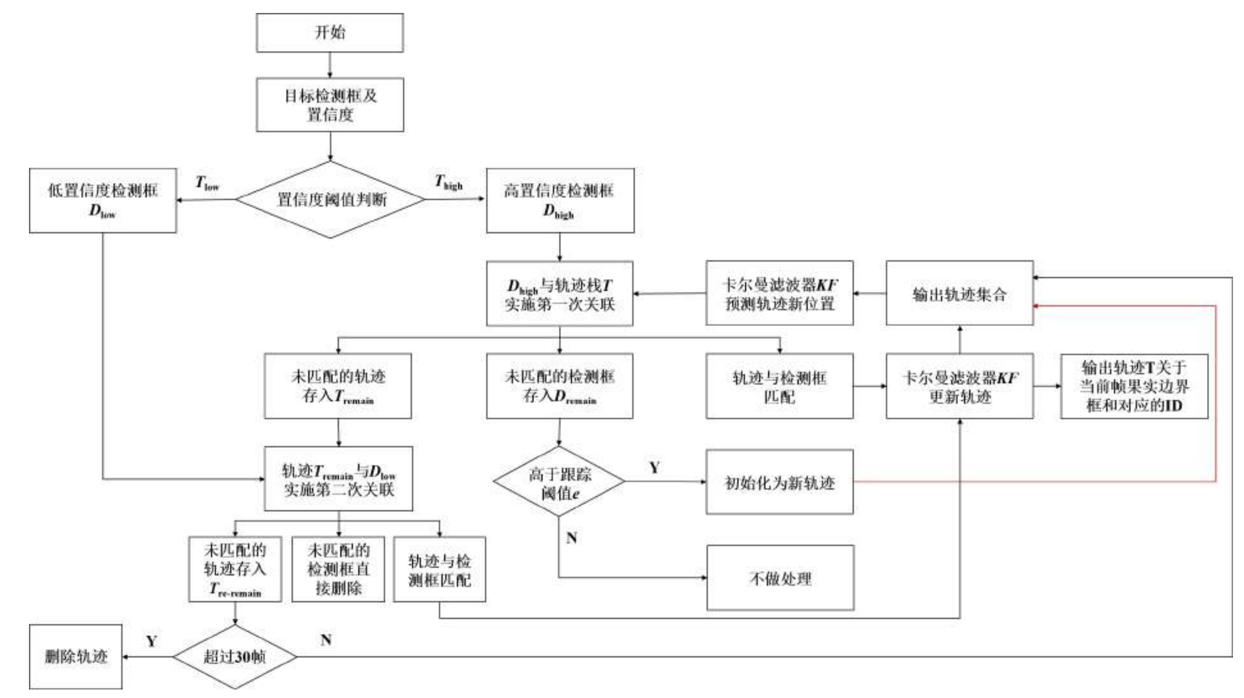


图6　柑橘果实跟踪流程图

Fig. 6 The flowchart for citrus fruit tracking

则从$T_{re\text{-}remain}$和T中删除，否则继续在T中保存。

（7）对于D_{remain}中的检测框，如果检测置信度高于e且存活超过两帧，则初始化为新的轨迹，并存入轨迹栈T。

（8）对于视频的每帧图像，输出T关于当前帧所有检测且跟踪到的果实边界框和对应的ID。

融合Byte数据关联算法的柑橘果实跟踪方法不仅在跟踪过程中贡献了必要的果实ReID信息，同时考虑了高分检测框和低分检测框的信息，有利于保持目标跟踪的连贯性。

3.4 损失函数

本研究所提方法采用的损失函数L_{total}由L_{heat}、L_{box}和$L_{identity}$的加权和获得，即公式（1）～（4）：

$$L_{heat}=-\frac{1}{N}\sum_{xy}\begin{cases}(1-\hat{M}_{xy})^{\alpha}\log(\hat{M}_{xy}), & M_{xy}=1\\ (1-M_{xy})^{\beta}(\hat{M}_{xy})^{\alpha}\log(1-\hat{M}_{xy}), & M_{xy}\neq 1\end{cases} \tag{1}$$

其中，$\hat{M}$为预测的heatmap特征图；M_{xy}为heatmap的真值；N是图中目标的总数量，个。

$$L_{box}=\sum_{i=1}^{N}\|\mathrm{o}^{i}-\hat{\mathrm{o}}^{i}\|_{1}+\lambda_{s}\|\mathrm{s}^{i}-\hat{\mathrm{s}}^{i}\|_{1}. \tag{2}$$

其中，ŝ和s分别指目标框大小的预测值与真值；ô和o分别指Offset中心点偏差的预测值与真值。

$$L_{identity}=-\sum_{i=1}^{N}\sum_{k=1}^{K}L^{i}(k)\log(\mathrm{p}(k)) \tag{3}$$

其中，p(k)指第k个物体的预测ID的可能性分布；$L(k)$指第k个物体真实的one-hot编码。

$$L_{total}=\frac{1}{2}\left(\frac{1}{e^{\omega_1}}(L_{heat}+L_{box})+\frac{1}{e^{\omega_2}}L_{identity}+\omega_1+\omega_2\right) \tag{4}$$

其中，ω_1和ω_2分别为目标检测损失和身份识别损失的可学习权重参数。

3.5 试验设置

为提高模型训练效率，试验中采用图形工作站进行模型训练。用于模型训练的硬件环境为：Intel i7-10700（2.90 GHz）CPU、NVIDIA GeForce RTX 3080（12 GB）GPU和32 GB RAM；训练用的软件环境为：Ubuntu 20.04 LTS OS，CUDA 11.6，Python 3.8和Pytorch 1.12深度学习框架。训练好的模型会移植到NVIDIA Jetson AGX边缘计算设备上进行性能测试。柑橘果实检测试验中的模型训练参数如表1所示。

表1 柑橘果实检测模型训练参数

Table 1 Parameters for models training in citrus fruit detection experiment

名称	数值
优化器	Adam
学习率	0.0005
Batch size	8
输入图像像素尺寸	608×1088
Num of workers	8
ReID dim	64
Max epoches	300

3.6 试验评价指标

对于果实检测试验，使用准确率（Precision，P）、召回率（Recall，R）、调和平均数（F_1 score，F_1）来评价模型目标检测性能，计算如公式（5）～（7）所示。同时，使用模型大小、参数量、浮点运算数计算量（Floating Point Operations，FLOPs）来评价模型复杂度。

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{5}$$

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{6}$$

$$F_1=\frac{2\times P\times R}{P+R} \tag{7}$$

其中，TP（True Positive）为检测正确的柑橘数，个；FP（False Positive）为误检测为柑橘的背景目标数，个；FN（False Negative）为误检测为背景的柑橘数，个。

对于果树挂果量估测试验，使用平均估测精度（Average Estimating Precision，AEP）来评价算法对视频中果树挂果量的估测精度，计算如公式（8）所示。

$$\mathrm{AEP}=\frac{\sum_{1}^{n}\left(1-\frac{|S-G|}{G}\right)}{n_{v}} \tag{8}$$

其中，S表示算法估测的柑橘数，个；G表示人工核实的柑橘实际数，个；n_v是用于测试的视频个数，个。

此外，使用决定系数 R^2 和均方根误差（Root Mean Square Error，RMSE）来评价算法估测果实数量与人工测得果实数量的比较情况。决定系数 R^2 用于衡量算法估测果实数量和人工测得果实数量之间的相关程度，数值越大表明二者相关性越好；RMSE用来衡量算法估测果实数量和人工测得果实数量之间的误差，其值越小表明误差越小。

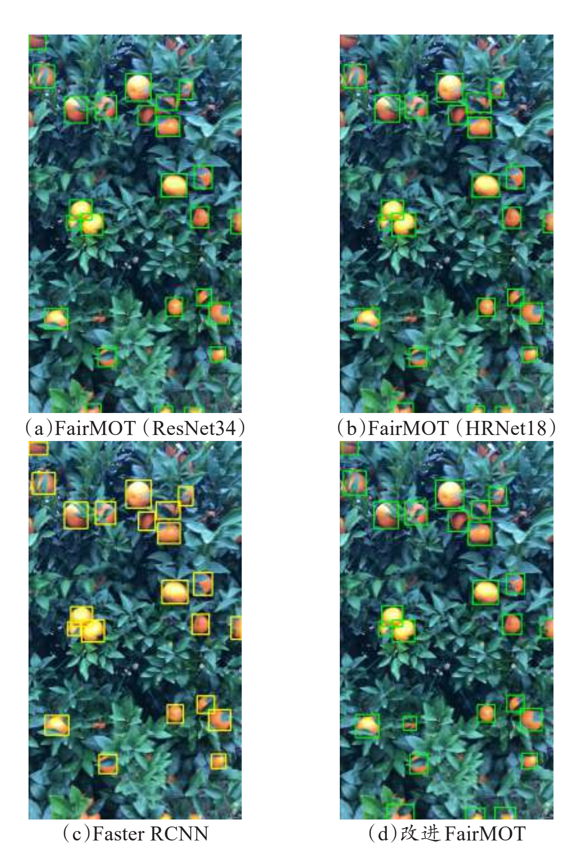
4 试验结果与分析

4.1 果实目标检测试验

4.1.1 不同模型柑橘果实检测性能对比

为验证本研究所建模型的有效性，对比试验中选择原始FairMOT框架使用的ResNet34、HRNet18作为特征提取网络与本研究采用CSPDarkNet53+PAFPN作为特征提取网络的改进FairMOT模型进行对比，并且增加Faster RCNN检测模型进行更为广泛的比较，评价果实目标检测性能以及模型复杂度。对比试验在相同的测试数据集下进行，测试集包含211幅柑橘树图像。果实检测效果如图7所示。由图7可以看出，相较于采用FairMOT（ResNet34）模型、FairMOT（HRNet18）模型和Faster RCNN模型，本研究所建的改进FairMOT模型可以更好地完成图像中柑橘果实的检测任务。

表2给出了不同模型果实检测性能的定量对比结果。由表2可知，采用本研究所建模型的CSP-DarkNet53+PAFPN结构，柑橘检测的准确率、召回率、调和平均数分别达到83.6%、89.2%、86.3%，性能均优于采用ResNet34结构和HRNet18结构的同类指标。Faster RCNN的柑橘检测准确率、召回率和调和平均指数分别为65.1%、69.0%和67.0%，明显低于所建模型的同类指标。试验结果表明，本研究所建模型采用的CSPDarkNet53+PAFPN结构能够更好地检测出果实位置，为树上柑橘挂果量估测的实现奠定基础。由于柑橘树枝叶茂密，会产生果实遮挡情况，可能造成算法在柑橘检测中的失误，但视频中同一果实会在视频的多个图像帧中出现，因此在某一帧因遮挡没有被检测到的果实，可以在后续视频帧中被检测到，从而在一定程度上避免对果树挂果量估测准确性的影响。



（a）FairMOT（ResNet34） （b）FairMOT（HRNet18） （c）Faster RCNN （d）改进FairMOT

图7 不同模型柑橘果实检测效果对比

Fig. 7 Comparison of citrus fruit detection effect of different models

表2 柑橘果实检测性能对比

Table 2 Comparison of citrus fruit detection performance

模型框架	特征提取网络	准确率/%	召回率/%	调和平均数/%
Faster RCNN	ResNet34	65.1	69.0	67.0
FairMOT	ResNet34	83.5	86.2	84.8
	HRNet18	83.1	89.1	86.0
改进FairMOT	CSPDark-Net53+PAFPN	83.6	89.2	86.3

4.1.2 模型复杂度对比

表3展示了不同模型的复杂度对比结果。由表3可知，采用本研究引入的CSPDarkNet53+PAF-PN结构后，模型参数量仅为采用ResNet34结构的20.19%和采用HRNet结构的41.51%，在FLOPs计算量方面比采用ResNet34结构和HRNet18结构分别少78.31%和87.63%，在模型大小方面仅为采用ResNet34结构和采用HRNet18结构的23.96%和45.00%；与Faster RCNN相比，本研究所建模型在参数量、FLOPs和模型大小等指标方面均具有明显优势。较低的模型复杂度证明本研究所采用的

CSPDarkNet53+PAFPN结构，对硬件计算资源有限的边缘计算设备更加友好，适合搭载到果园移动平台完成高效果实作业任务。

表3 不同模型的复杂度对比

Table 3 Comparison of different models' complexities

模型框架	特征提取网络	参数量/M	FLOPs/G	模型大小/MB
Faster RCNN	ResNet34	38.39	114.96	153.7
FairMOT	ResNet34	24.82	168.00	293.0
	HRNet18	12.07	294.62	156.0
改进FairMOT	CSPDarkNet53+PAFPN	5.01	36.44	70.2

4.1.3 不同轻量主干网络的消融试验

为验证CSPDarkNet53主干网络的有效性以及PAFPN对模型精度和模型大小的影响，进一步开展了消融实验。选择主流轻量网络EfficientNet-Lite与CSPDarkNet53进行比较，试验结果如表4所示。

表4 不同轻量主干网络的消融试验结果

Table 4 Ablation results of different lightweight backbone networks

主干网络	PAFPN	准确率/%	召回率/%	调和平均数/%	参数量/M	FLOPs/G	模型大小/MB
EfficientNet-Lite		83.0	85.3	84.1	8.24	101.28	109.1
	√	83.2	86.6	84.9	8.29	101.96	109.8
CSPDarkNet53		83.3	88.7	85.9	4.93	35.42	69.1
	√	83.6	89.2	86.3	5.01	36.44	70.2

由表4可知，相比于EfficientNet-Lite，采用CSPDarkNet53的果实检测性能和模型复杂度均表现出更加优秀的性能。在不使用PAFPN的情况下，采用CSPDarkNet53作为主干网络对柑橘检测的准确率、召回率、调和平均数分别为83.3%、88.7%、85.9%，模型参数量、FLOPs和模型大小分别为4.93 M、35.42 G和69.1 MB；在CSPDarkNet53后使用了PAFPN的柑橘检测准确率、召回率、调和平均数均得到了提升，且模型大小基本相同，证明了使用PAFPN对于模型性能提升具有积极的影响。

4.2 果树挂果量估测试验

4.2.1 不同模型挂果量估测性能对比

为验证本研究所建模型在果树挂果量估测任务中的有效性，同时便于后续实际果园工况条件下的应用，研发了果园挂果量远程监测系统，将算法模型嵌入监测系统，实现果树挂果量的自动估测。其中，集成了挂果量估测模型的监测系统服务端部署在边缘计算设备上，用于挂果量自动估测，监测系统监控端部署在普通电脑中，用于接收和显示挂果量估测结果。果园挂果量远程监测系统运行效果如图8所示。



(a)服务端



(b)监控端

图8 柑橘果园挂果量远程监测系统运行图

Fig. 8 Operation diagram of the fruit load remote monitoring system of citrus orchard

试验中，随机选用10个测试视频，包含10棵柑橘树，视频中人工核实的柑橘实际数分别为39、70、125、14、81、15、47、32、27和63个。利用

测试视频对训练得到的模型分别进行挂果量估测，评价不同特征提取网络和跟踪策略对于挂果量估测的精度和速度性能，结果如表5所示。

表5 不同模型的挂果量估测性能对比

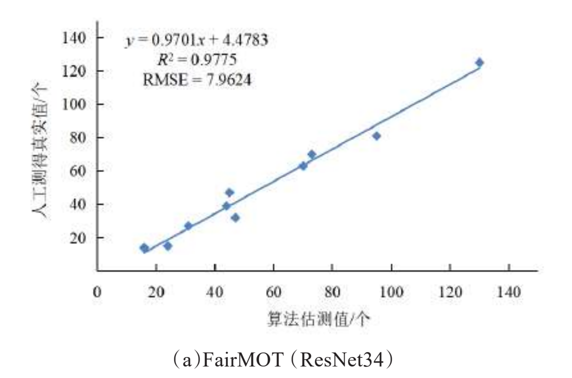
Table 5 Performance comparison of different models for fruit load estimation

模型框架	特征提取网络	跟踪策略	AEP/%	处理速度/(f·s^{-1})
FairMOT	ResNet34	改进前	85.67	6.02
		改进后	86.98	6.05
	HRNet18	改进前	90.75	3.12
		改进后	91.53	3.14
改进FairMOT	CSPDarkNet53+PAFPN	改进前	85.76	14.48
		改进后	91.61	14.76

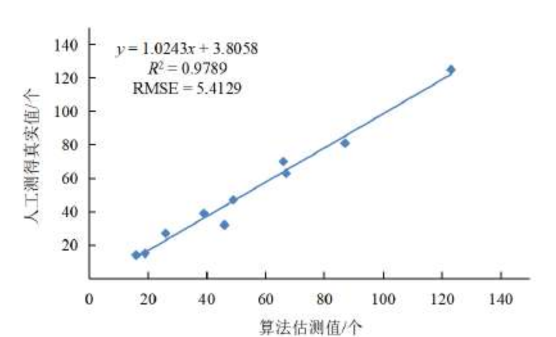
由表5可以看出，在FairMOT的基础上融合Byte算法得到的改进跟踪策略，对于提升果实挂果量的估测精度产生了正面效果。特征提取网络ResNet34、HRNet18和CSPDarkNet53+PAFPN再结合改进后的跟踪策略所得模型，AEP分别达到86.98%、91.53%和91.61%，处理速度分别达到6.05、3.14和14.76 f/s。相较于采用ResNet34结构和HRNet18结构，本研究所建模型采用的CSP-DarkNet53+PAFPN结构结合改进后的果实跟踪策略，表现出了更优的挂果量估测性能，在保持较高估测精度的同时，处理速度分别是对比模型的2.4倍和4.7倍，证明了本研究所建模型在边缘计算设备上对柑橘挂果量估测的有效性。

4.2.2 柑橘挂果量估测定量化数值拟合分析

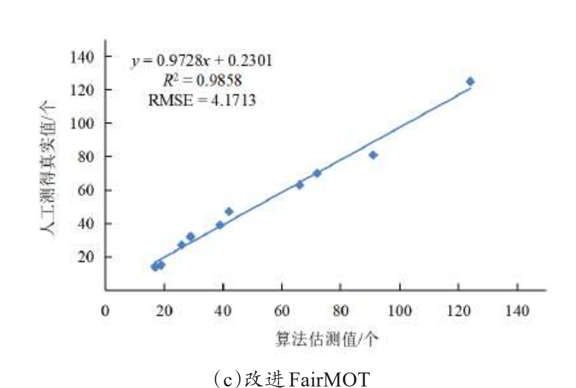
为进一步分析所建模型在果树挂果量估测中的性能，对挂果量算法估测值与人工测得真实值进行定量化数值拟合分析。图9展示了柑橘挂果量估测数值拟合结果对比。由图9可知，本研究所建的改进FairMOT模型测得挂果量与人工测得真实值之间的RMSE为4.1713，比FairMOT（ResNet34）和FairMOT（HRNet18）分别低47.61%和22.94%。通过观察拟合结果可知，本研究所建模型测得结果值与人工测得真实值的决定系数R^2为0.9858，优于其他两种对比模型，表明所建模型对视频中柑橘挂果量估测值与人工测得真实值具有更好相关性。



(a)FairMOT（ResNet34）



(b)FairMOT(HRNet18)



(c)改进FairMOT

图9 挂果量算法估测值与人工测得真实值拟合结果

Fig. 9 Fitting results of the fruit loads from algorithm estimation and the ground truth measured manually

5 结 论

为实现果园工况条件下果树挂果量的自动估测，本研究提出一种用于边缘计算设备的轻量化模型，实现视频中树上柑橘挂果量的自动估测。

（1）模型采用CSPDarkNet53+PAFPN结构作为特征提取网络，在保证特征表征能力的前提下实现更快的推理速度和更低的模型复杂度，在果实跟踪阶段引入Byte算法改进FairMOT的数据关联策略，对视频中柑橘果实进行预测跟踪，以提升挂果量估测准确性。

（2）将视频数据作为输入，利用模型自动估测树上挂果量，本研究所建模型的平均估测精度（AEP）和处理速度分别达到91.61%和14.76 f/s，测得结果值与真实值的决定系数R^2为0.9858，均方根误差（RMSE）为4.1713，模型参数量、FLOPs和模型大小分别为5.01 M、36.44 G和70.2 MB，展现出较对比模型更优的挂果量估测性能和更低的模型复杂度，证明了本研究所提方法在边缘计算设备上对柑橘挂果量估测的有效性。

（3）基于算法模型研发的果园挂果量远程监测系统，用于满足果园移动平台行进状态下的果树挂果量监测需求，可为果树生产力自动监测分析提供技术支持，在智慧果园生产中具有良好应用前景。未来研究将继续丰富数据资源，进一步改进模型性能，探索更加高效的方法以满足更多果树品种的挂果量估测需求。

利益冲突声明：本研究不存在研究者以及与公开研究成果有关的利益冲突。

参考文献：

[1] FENG A J, ZHOU J F, VORIES E D, et al. Yield estimation in cotton using UAV-based multi-sensor imagery[J]. Biosystems engineering, 2020, 193: 101-114.

[2] KURTULMUS F, LEE W S, VARDAR A. Green citrus detection using 'eigenfruit', color and circular Gabor texture features under natural outdoor conditions[J]. Computers and electronics in agriculture, 2011, 78(2): 140-149.

[3] QURESHI W S, PAYNE A, WALSH K B, et al. Machine vision for counting fruit on mango tree canopies[J]. Precision agriculture, 2017, 18(2): 224-244.

[4] ZHOU R, DAMEROW L, SUN Y R, et al. Using colour features of cv. 'Gala' apple fruits in an orchard in image processing to predict yield[J]. Precision agriculture, 2012, 13(5): 568-580.

[5] ANNAMALAI P, LEE W S. Citrus yield mapping system using machine vision[C]// 2003, Las Vegas, NV July 27-30, 2003. St. Joseph, MI, USA: American Society of Agricultural and Biological Engineers, 2003: 1.

[6] STAJNKO D, RAKUN J, BLANKE M. Modelling apple fruit yield using image analysis for fruit colour, shape and texture[J]. European journal of horticultural science, 2009, 74(6): 260-267.

[7] DORJ U O, LEE M, YUN S S. An yield estimation in citrus orchards via fruit detection and counting using image processing[J]. Computers and electronics in agriculture, 2017, 140: 103-112.

[8] SA I, GE Z Y, DAYOUB F, et al. DeepFruits: A fruit detection system using deep neural networks[J]. Sensors, 2016, 16(8): ID 1222.

[9] CHEN S W, SHIVAKUMAR S S, DCUNHA S, et al. Counting apples and oranges with deep learning: A data-driven approach[J]. IEEE robotics and automation letters, 2017, 2(2): 781-788.

[10] BARGOTI S, UNDERWOOD J. Deep fruit detection in orchards[C]// 2017 IEEE International Conference on Robotics and Automation (ICRA). Piscataway, NJ, USA: IEEE, 2017: 3626-3633.

[11] HÄNI N, ROY P, ISLER V. A comparative study of fruit detection and counting methods for yield mapping in apple orchards[J]. Journal of field robotics, 2020, 37(2): 263-282.

[12] 李志军, 杨圣慧, 史德帅, 等. 基于轻量化改进YOLOv5的苹果树产量测定方法[J]. 智慧农业(中英文), 2021, 3(2): 100-114.
LI Z J, YANG S H, SHI D S, et al. Yield estimation method of apple tree based on improved lightweight YOLOv5[J]. Smart agriculture, 2021, 3(2): 100-114.

[13] KESTUR R, MEDURI A, NARASIPURA O. MangoNet: A deep semantic segmentation architecture for a method to detect and count mangoes in an open orchard[J]. Engineering applications of artificial intelligence, 2019, 77: 59-69.

[14] 高芳芳, 武振超, 索睿, 等. 基于深度学习与目标跟踪的苹果检测与视频计数方法[J]. 农业工程学报, 2021, 37(21): 217-224.
GAO F F, WU Z C, SUO R, et al. Apple detection and counting using real-time video based on deep learning and object tracking[J]. Transactions of the Chinese society of agricultural engineering, 2021, 37(21): 217-224.

[15] WANG Z L, WALSH K, KOIRALA A. Mango fruit load estimation using a video based MangoYOLO-kalman filter-hungarian algorithm method[J]. Sensors, 2019, 19(12): ID 2742.

[16] LUO W H, XING J L, MILAN A, et al. Multiple object tracking: A literature review[J]. Artificial intelligence, 2021, 293: ID 103448.

[17] RAKAI L, SONG H S, SUN S J, et al. Data association in multiple object tracking: A survey of recent techniques[J]. Expert systems with applications, 2022, 192: ID 116300.

[18] 涂淑琴, 汤寅杰, 李承桀, 等. 基于改进ByteTrack算法的群养生猪行为识别与跟踪技术[J]. 农业机械学报, 2022, 53(12): 264-272.
TU S Q, TANG Y J, LI C J, et al. Behavior recognition and tracking of group-housed pigs based on improved ByteTrack algorithm[J]. Transactions of the Chinese society for agricultural machinery, 2022, 53(12): 264-272.

[19] ZHANG Y F, WANG C Y, WANG X G, et al. FairMOT: On the fairness of detection and re-identification in multiple object tracking[J]. International journal of computer vision, 2021, 129(11): 3069-3087.

[20] ZHANG Y F, SUN P Z, JIANG Y, et al. ByteTrack: Multi-object tracking by associating every detection box[C]// European conference on computer vision. Berlin, German: Springer, 2022: 1-21.

[21] 吴昊. 基于YOLOX和重识别的行人多目标跟踪方法[J]. 自动化与仪表, 2023, 38(3): 59-62, 67.
WU H. Pedestrian multi-target tracking method based on YOLOX and person re-identification[J]. Automation & instrumentation, 2023, 38(3): 59-62, 67.

[22] OUYANG W L, WANG X G, ZENG X Y, et al. DeepID-Net: Deformable deep convolutional neural networks for object detection[C]// 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2015: 2403-2412.

[23] REDMON J, FARHADI A. YOLOv3: An incremental improvement[EB/OL]. arXiv: 1804.02767, 2018.

[24] WANG C Y, MARK LIAO H Y, WU Y H, et al. CSPNet: A new backbone that can enhance learning capability of CNN[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Piscataway, NJ, USA: IEEE, 2020: 1571-1580.

[25] 韦锦, 李正强, 许恩永, 等. 基于DA2-YOLOv4算法绿篱识别研究[J]. 中国农机化学报, 2022, 43(9): 122-130.
WEI J, LI Z Q, XU E Y, et al. Research on hedge recognition based on DA2-YOLOv4 algorithm[J]. Journal of Chinese agricultural mechanization, 2022, 43(9): 122-130.

[26] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. Scaled-YOLOv4: Scaling cross stage partial network[C]// 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2021: 13024-13033.

[27] GÜNEY E, BAYILMIŞ C, ÇAKAN B. An implementation of real-time traffic signs and road objects detection based on mobile GPU platforms[J]. IEEE access, 2022, 10: 86191-86203.

[28] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2017: 936-944.

[29] LIU S, QI L, QIN H F, et al. Path aggregation network for instance segmentation[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE, 2018: 8759-8768.

[30] 孙泽强, 陈炳才, 崔晓博, 等. 融合频域注意力机制和解耦头的YOLOv5带钢表面缺陷检测[J]. 计算机应用, 2023, 43(1): 242-249.
SUN Z Q, CHEN B C, CUI X B, et al. Strip steel surface defect detection by YOLOv5 algorithm fusing frequency domain attention mechanism and decoupled head[J]. Journal of computer applications, 2023, 43(1): 242-249.

A Lightweight Fruit Load Estimation Model for Edge Computing Equipment

XIA Xue[1], CHAI Xiujuan[1], ZHANG Ning[1*], ZHOU Shuo[1], SUN Qixin[1], SUN Tan[2*]

（1. *Agricultural Information Institute*, *Chinese Academy of Agricultural Sciences/Key Laboratory of Agricultural Big Data*, *Ministry of Agriculture and Rural Affairs*, *Beijing* 100081, *China*; 2. *Chinese Academy of Agricultural Sciences*, *Beijing* 100081, *China*）

Abstract:

[Objective] The fruit load estimation of fruit tree is essential for horticulture management. Traditional estimation method by manual sampling is not only labor-intensive and time-consuming but also prone to errors. Most existing models can not apply to edge computing equipment with limited computing resources because of their high model complexity. This study aims to develop a lightweight model for edge computing equipment to estimate fruit load automatically in the orchard.

[Methods] The experimental data were captured using the smartphone in the citrus orchard in Jiangnan district, Nanning city, Guangxi province. In the dataset, 30 videos were randomly selected for model training and other 10 for testing. The general idea of the proposed algorithm was divided into two parts: Detecting fruits and extracting ReID features of fruits in each image from the video,

then tracking fruit and estimating the fruit load. Specifically, the CSPDarknet53 network was used as the backbone of the model to achieve feature extraction as it consumes less hardware computing resources, which was suitable for edge computing equipment. The path aggregation feature pyramid network PAFPN was introduced as the neck part for the feature fusion via the jump connection between the low-level and high-level features. The fused features from the PAFPN were fed into two parallel branches. One was the fruit detection branch and another was the identity embedding branch. The fruit detection branch consisted of three prediction heads, each of which performed 3×3 convolution and 1×1 convolution on the feature map output by the PAFPN to predict the fruit's keypoint heat map, local offset and bounding box size, respectively. The identity embedding branch distinguished between different fruit identity features. In the fruit tracking stage, the byte mechanism from the ByteTrack algorithm was introduced to improve the data association of the FairMOT method, enhancing the performance of fruit load estimation in the video. The Byte algorithm considered both high-score and low-score detection boxes to associate the fruit motion trajectory, then matches the identity features' similarity of fruits between frames. The number of fruit IDs whose tracking duration longer than five frames was counted as the amount of citrus fruit in the video.

[Results and Discussions] All experiments were conducted on edge computing equipment. The fruit detection experiment was conducted under the same test dataset containing 211 citrus tree images. The experimental results showed that applying CSPDarkNet53+ PAFPN structure in the proposed model achieved a precision of 83.6%, recall of 89.2% and F_1 score of 86.3%, respectively, which were superior to the same indexes of FairMOT (ResNet34) model, FairMOT (HRNet18) model and Faster RCNN model. The CSP-DarkNet53+PAFPN structure adopted in the proposed model could better detect the fruits in the images, laying a foundation for estimating the amount of citrus fruit on trees. The model complexity experimental results showed that the number of parameters, FLOPs (Floating Point Operations) and size of the proposed model were 5.01 M, 36.44 G and 70.2 MB, respectively. The number of parameters for the proposed model was 20.19% of FairMOT (ResNet34) model's and 41.51% of FairMOT (HRNet18) model's. The FLOPs for the proposed model was 78.31% less than FairMOT (ResNet34) model's and 87.63% less than FairMOT (HRNet18) model's. The model size for the proposed model was 23.96% of FairMOT (ResNet34) model's and 45.00% of FairMOT (HRNet18) model's. Compared with the Faster RCNN, the model built in this study showed advantages in the number of parameters, FLOPs and model size. The low complexity proved that the proposed model was more friendly to edge computing equipment. Compared with the lightweight backbone network EfficientNet-Lite, the CSPDarkNet53 applied in the proposed model's backbone performed better fruit detection and model complexity. For fruit load estimation, the improved tracking strategy that integrated the Byte algorithm into the FairMOT positively boosted the estimation accuracy of fruit load. The experimental results on the test videos showed that the AEP (Average Estimating Precision) and FPS (Frames Per Second) of the proposed model reached 91.61% and 14.76 f/s, which indicated that the proposed model could maintain high estimation accuracy while the FPS was 2.4 times and 4.7 times of the comparison models, respectively. The RMSE (Root Mean Square Error) of the proposed model was 4.1713, which was 47.61% less than FairMOT (ResNet34) model's and 22.94% less than FairMOT (HRNet18) model's. The R^2 of the determination coefficient between the algorithm-measured value and the manual counted value was 0.9858, which was superior to other comparison models. The proposed model revealed better performance in estimating fruit load and lower model complexity than other comparatives.

[Conclusions] The experimental results proved the validity of the proposed model for fruit load estimation on edge computing equipment. This research could provide technical references for the automatic monitoring and analysis of orchard productivity. Future research will continue to enrich the data resources, further improve the model's performance, and explore more efficient methods to serve more fruit tree varieties.

Key words: smart orchard; fruit load estimation; edge computing; deep learning; multiple object tracking; lightweight model